Decentralized Inventory Control for Large-Scale Reverse Supply Chains: A Computationally Tractable Approach

Prashanth Krishnamurthy, *Member, IEEE*, Farshad Khorrami, *Senior Member, IEEE*, and David Schoenwald, *Senior Member, IEEE*

***Abstract*—In this paper, we consider a new inventory control technique for large-scale supply chains including repairs. The part flow is bidirectional with broken parts propagated upstream for repair. It is well known that available optimization techniques for inventory control for bidirectional stochastic supply chains are computationally intractable and also necessitate several simplifying assumptions. In contrast, the proposed approach is an adaptive scheme that scales well to practically interesting large-scale multi-item supply chains. Furthermore, practical issues such as stochastic transport delays, manufacturing times, and repair times and probabilistic characterization of part repair success are handled in a unified framework. The control scheme is based on a hierarchical two-level architecture that comprises an adaptive set point generator and a lower-level order-up-to policy. An application to aircraft supply chains involving multiple original equipment manufacturers (OEMs), depots, bases, squadrons, and planes is also investigated.**

***Index Terms*—Decentralized control, inventory control, large-scale systems, reverse supply chain.**

I. Introduction

INVENTORY control for large-scale supply chains is well recognized [1]–[3] as an important problem with numerous applications including manufacturing systems, logistics systems, communication networks, and transportation systems. Considerable work on both modeling and control of supply chains has been reported in the literature. A review and literature survey of supply chain modeling techniques can be found in [4]. The existing results on inventory control for supply chains focus primarily on single-directional supply chains [5]–[13], wherein parts flow from manufacturers to end users through a chain of transportation and storage nodes. In this case, fairly general results have been obtained, especially in the case when the supply chain consists of only one supplier and one client [7], [10]. However, these results rely crucially on the assumption that the part flow is single directional and cannot be extended to bidirectional part flow.

In recent years, bidirectional supply chains (or *reverse* supply chains) have attained increasing importance [14]–[16], especially in two contexts, one being the case of supply chains that also handle repairs (as is typical in any *maintenance* supply chain) and the second being the case of supply chains that include recycling, whether for environmental or economic reasons. Unlike single-directional supply chains, optimization-based approaches to bidirectional supply chains are computationally intractable for realistic supply chains (partly owing to the property that stochastic disturbances enter at both ends of a bidirectional supply chain), and also necessitate simplifying assumptions on manufacturing times, repair times, demand profiles, etc. In this paper, we propose a new inventory control technique for large-scale bidirectional supply chains. The control scheme is based on a hierarchical two-level architecture that is obtained through a novel formulation of a bidirectional supply chain, and the control objective that is framed in an inherently decentralized setting. The higher level controller in the hierarchical two-level architecture is an adaptive inventory set point generator that performs online tuning of the desired inventory levels, while the lower level controller follows an order-up-to policy. The controller is of a very simple structure, and is computationally tractable even for very large-scale supply chains. Furthermore, the applicability of the proposed scheme is enhanced through a decentralized approach. We provide both a fully decentralized scheme and a partially decentralized scheme (wherein each site communicates with its neighbors in the supply chain).

A mathematical model for the class of supply chains considered is developed in Section II. The proposed inventory control strategy is provided in Section III. In Section IV, we consider the application of the obtained results to aircraft supply chains [17], [18], which form a challenging and important example of large-scale supply chains. A general-purpose simulation package supporting supply chains with arbitrary numbers of nodes, part types, and demand characteristics is presented in Section V. Simulation studies demonstrating the performance of the proposed inventory controllers are also provided in Section V.

Manuscript received June 23, 2006; revised July 9, 2007. This work was supported in part by the Laboratory Directed Research and Development (LDRD) Program at Sandia National Laboratories. Sandia National Laboratories is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the U.S. Department of Energy under Contract DE-AC04-94AL85000. An earlier version of this paper was presented at the 2006 American Control Conference, Minneapolies, MN. This paper was recommended by Editor V. Marik.

P. Krishnamurthy is with the IntelliTech Microsystems, Inc., Bowie, MD 20715 USA (e-mail: pkrishnamurthy@imicro.biz).

F. Khorrami is with the Control and Robotics Research Laboratory (CRRL), Department of Electrical and Computer Engineering, Polytechnic University, Brookly, NY 11201, USA (e-mail: khorrami@smart.poly.edu).

D. Schoenwald is with the Sandia National Laboratories, Albuquerque, NM 87185 USA (e-mail: daschoe@sandia.gov).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMCC.2007.913906

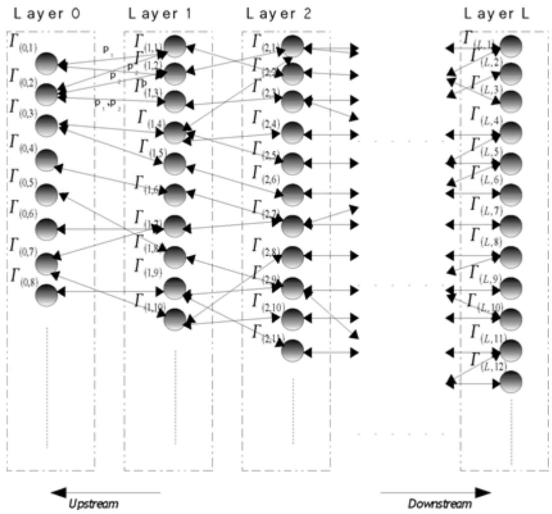

Fig. 1. Supply chain network.

II. Modeling Considerations for Supply Chains With Repairs

We consider a general supply chain composed of manufacturing sites, repair sites, and client sites. Supply chains of the form considered appear in various applications and encompass the category of support networks wherein the purpose of the supply chain is to provide repair and manufacturing services to a set of *end-nodes* that are required to satisfy some performance criteria. Typically, such networks have some sites that are responsible for manufacturing new parts and several intermediate sites that maintain inventories and possess some repair capabilities. A mathematical framework for such supply chains is developed next.

The supply chain is modeled as a network of sites $\Gamma_{(i,j)}, i = 0,\ldots,L, j = 1,\ldots,N_i$ where $L, N_0,\ldots,N_L$ are positive constants. The network is organized as being made of $L+1$ layers (see Fig. 1) with $\Gamma_{(i,j)}, j = 1,\ldots,N_i$ forming the ith layer. The sites $\Gamma_{(0,j)}, j = 1,\ldots,N_0$ that form the 0th layer are the *manufacturing sites*. The sites $\Gamma_{(i,j)}, i = 1,\ldots, L-1, j = 1,\ldots,N_i$ are the *intermediate sites*, while the sites $\Gamma_{(L,j)}, j = 1,\ldots,N_L$ are the *end-nodes*. As indicated in Fig. 1, the number of sites N_i at layer i generally increases with i. Both the manufacturing sites $\Gamma_{(0,j)}, j = 1,\ldots,N_0$ and the intermediate sites $\Gamma_{(i,j)}, i = 1,\ldots,L-1, j = 1,\ldots,N_i$ possess repair capabilities, though to varying degrees, as modeled by associated probabilities. Only the manufacturing sites $\Gamma_{(0,j)}, j = 1,\ldots,N_0$ possess manufacturing capabilities. The end-nodes $\Gamma_{(L,j)}, j = 1,\ldots,N_L$ possess neither repair nor manufacturing capabilities.[1] Inventory stocks are held at the sites $\Gamma_{(i,j)}, i = 0,\ldots,L-1, j = 1,\ldots,N_i$ and parts utilized at the end-nodes $\Gamma_{(L,j)}, j = 1,\ldots,N_L$. The performance criterion is formulated in terms of the parts available at the end-nodes. A typical example for the outlined class of supply chains is an aircraft supply chain, wherein a set of aircraft form the end-nodes and require a certain set of parts to be mission-capable. The application of the proposed inventory control technique to an aircraft supply chain is considered in Section IV. A variety of other supply chains including equipment or machinery support networks also fall within the class of supply chains considered.

The part types handled by the supply chain are denoted by $p_1,\ldots,p_P$, with P being the number of different part types. The inventory stock of part p_k at site $\Gamma_{(i,j)}$ at time t is denoted by $\Gamma^{p_k}_{(i,j)}(t)$. The supply chain is bidirectional in the sense that parts can flow in either direction (left to right or right to left), as shown in Fig. 1. New parts are manufactured at the sites $\Gamma_{(0,j)}, j = 1,\ldots,N_0$ and propagate *downstream* toward the end-nodes, i.e., from left to right in Fig. 1. Parts are utilized at the end-nodes and fail after a random duration of time[2] characterized by a probability density function after which the broken part is propagated *upstream*, i.e., to the left in Fig. 1. Each site that receives a broken part attempts to repair it. If successful, the repaired part is sent back downstream, i.e., to the right in Fig. 1. If the repair attempt is unsuccessful, however, the broken part is propagated one level further upstream. This continues till a manufacturing site receives the broken part. If the manufacturing site is unsuccessful in repairing the broken part, then the broken part is discarded and a new part is manufactured in its place. We focus on a *pull* strategy throughout, wherein any site ships a part downstream or initiates a repair attempt only when a downstream site explicitly requests it. This is in keeping with the "inventory is waste" and just-in-time (JIT) philosophies [3].

Each of the end-nodes has an associated set of required parts for the end-node to be considered functional. In general, the part requirements at the end-nodes could be quite complex and involve alternative parts, optional parts, etc. For simplicity, we consider a scenario wherein each of the end-nodes $\Gamma_{(L,j)}$ has an associated set $\{n^{p_k}_{(L,j)}: k = 1,\ldots,P\}, j = 1,\ldots,N_L$, specifying the numbers of parts of each part type p_k required at site $\Gamma_{(L,j)}$. While for simplicity in this paper, we focus on the case that $n^{p_k}_{(L,j)}, k = 1,\ldots,P, j = 1,\ldots,N_L$ are constants, the proposed inventory control algorithms can, due to their adaptive nature, handle part requirements that vary with time, i.e., $n^{p_k}_{(L,j)}$ could be functions of time. Further generalizations for more complex part requirements can be developed along similar lines as in this paper; the details are omitted for brevity.

For each part type, each site has a designated supplier site at the next higher level. The site that acts as the supplier of part p_k to site $\Gamma_{(i,j)}$ is denoted by $\mathcal{S}(\Gamma_{(i,j)};p_k)$. Formally, for each $k \in \{1,\ldots,P\}$, the following is true for each $i \in \{1,\ldots,L\}$ and $j \in \{1,\ldots,N_i\}$:

$$\exists \text{ a unique } m \in \{1,\ldots,N_{i-1}\}$$
$$\text{such that } \mathcal{S}(\Gamma_{(i,j)};p_k) = \Gamma_{(i-1,m)}. \tag{1}$$

[1] This characterization of end-nodes is introduced for simplicity in demarcating the roles of the nodes, and can be easily relaxed by extending the supply chain to include an additional conceptual layer of virtual end-nodes, with the transportation delay between the actual end-nodes and the virtual end-nodes being zero.

[2] For simplicity, it is assumed that parts only fail at the end-nodes. Practically, this implies that the shelf life must be much larger than the mean time before failure during active use. In the case that this assumption is not satisfied, the controller design and analysis can be extended by appropriately modifying the inventory deficit signal.

To denote the set of sites for which a given site acts as a supplier for a given part, we introduce the notation

$$\mathcal{S}^{-1}(\Gamma_{(i,m)}; p_k) = \{j \in \{1, \ldots, N_{i+1}\} \mid \mathcal{S}(\Gamma_{(i+1,j)}; p_k) = \Gamma_{(i,m)}\}. \quad (2)$$

From the definitions of $\mathcal{S}$ and $\mathcal{S}^{-1}$, we have the relationship

$$\bigcup_{j=1}^{N_i} \mathcal{S}^{-1}(\Gamma_{(i,j)}; p_k) = \{1, \ldots, N_{i+1}\} \quad (3)$$

valid for all $i \in \{0, \ldots, L-1\}$ and $k \in \{1, \ldots, P\}$. Moreover, the union on the left-hand side of (3) is a disjoint union.

Note that the suppliers are defined partwise. This takes into account practical scenarios with different suppliers for different parts. Furthermore, the adaptive controller developed in Section III can handle dynamic supplier relationships, i.e., wherein the supplier $\mathcal{S}(\Gamma_{(i,j)}; p_k)$ is time-dependent,[3] possibly for each i, j, and k. The adaptive performance of the proposed controller in the face of changing supplier relationships can be seen in a simulation example in Section V.

We next describe the behavior of each site. For convenience, we utilize a discrete time base, $t_i, i = 0, 1, 2, \ldots$ with the events occurring in the time interval $(t_{n-1}, t_n]$ assumed for the purpose of modeling and control design, to occur at the time t_n. Note that the underlying supply chain system is asynchronous and event-driven. Hence, the time-differential $\Delta t = (t_n - t_{n-1})$ should be picked appropriately, depending on the particular application to minimize controller action delays and to ensure simulation fidelity. The time differential can be typically taken to be of the order of 1 day for aircraft supply chains. Smaller time differentials ($\sim$1 h) are also used for high-activity aircraft supply chains. At time t_n, the events that can occur at a site $\Gamma_{(i,j)}, i = 0, \ldots, L-1, j = 1, \ldots, N_i$ and the resulting actions are as follows.

1) *A broken part of type p_k is received from a site $\Gamma_{(i+1,m)} \in \{\Gamma_{(i+1,r)} \mid r \in \mathcal{S}^{-1}(\Gamma_{(i,j)}; p_k)\}$:* In this case, a repair attempt for the broken part is initiated. Also, if a working part of type p_k is currently in the on-site inventory at site $\Gamma_{(i,j)}$, then it is sent to the site $\Gamma_{(i+1,m)}$. On the other hand, if a working part of type p_k is not currently available in the on-site inventory at site $\Gamma_{(i,j)}$, then $\Gamma_{(i+1,m)}$ is added to the end of a list of outstanding orders $\mathcal{O}(\Gamma_{(i,j)}; p_k)$ of type p_k maintained at site $\Gamma_{(i,j)}$.
2) *A repair attempt of a part of type p_k completes successfully:* If $\mathcal{O}(\Gamma_{(i,j)}; p_k)$ is empty, then the repaired part is added to the on-site inventory. On the other hand, if $\mathcal{O}(\Gamma_{(i,j)}; p_k)$ is nonempty, the repaired part is sent to the first site in $\mathcal{O}(\Gamma_{(i,j)}; p_k)$, and the first entry in $\mathcal{O}(\Gamma_{(i,j)}; p_k)$ is removed.
3) *A repair attempt of a part of type p_k completes unsuccessfully:* If the site is an intermediate site, i.e., if $i \in \{1, \ldots, L-1\}$, then the part is sent to $\mathcal{S}(\Gamma_{(i,j)}; p_k)$. On the other hand, if the site is a manufacturing site, i.e., if $i = 0$, then the part is discarded.
4) *(Only relevant for an intermediate site, i.e., if $i \in \{1, \ldots, L-1\}$) A working part of type p_k is received from site $\mathcal{S}(\Gamma_{(i,j)}; p_k)$:* If $\mathcal{O}(\Gamma_{(i,j)}; p_k)$ is empty, then the part is added to the on-site inventory. If $\mathcal{O}(\Gamma_{(i,j)}; p_k)$ is nonempty, the part is sent to the first site in $\mathcal{O}(\Gamma_{(i,j)}; p_k)$, and the first entry in $\mathcal{O}(\Gamma_{(i,j)}; p_k)$ is removed.
5) *(Only relevant for a manufacturing site, i.e., if $i = 0$) The manufacture of a part of type p_k completes:* If $\mathcal{O}(\Gamma_{(0,j)}; p_k)$ is empty, then the new part is added to the on-site inventory. If $\mathcal{O}(\Gamma_{(0,j)}; p_k)$ is nonempty, then the part is sent to the first site in $\mathcal{O}(\Gamma_{(0,j)}; p_k)$ and the first entry in $\mathcal{O}(\Gamma_{(0,j)}; p_k)$ is removed.

The events that can occur at one of the end-nodes $\Gamma_{(L,j)}, j = 1, \ldots, N_L$ at time t_n and the resulting actions are as follows:

1) *A part of type p_k fails*: The failed part is sent to the site $\mathcal{S}(\Gamma_{(L,j)}; p_k)$.
2) *A working part of type p_k is received from site $\mathcal{S}(\Gamma_{(L,j)}; p_k)$*: The part is added to the on-site inventory.

The amount of time required for a part to travel from one site to another is characterized via probability distributions defined for each pair $(\Gamma_{(i,j)}, \mathcal{S}(\Gamma_{(i,j)}; p_k)), i = 1, \ldots, L, j = 1, \ldots, N_i, k = 1, \ldots, P$. The probability distribution governing the amount of time taken for a part to move from $\Gamma_{(i,j)}$ to $\mathcal{S}(\Gamma_{(i,j)}; p_k)$ need not be the same as the probability distribution governing the amount of time taken for a part to move from $\mathcal{S}(\Gamma_{(i,j)}; p_k)$ to $\Gamma_{(i,j)}$. The amounts of time required for repair attempts and part manufactures are also, in general, governed by probability distributions defined for each part and site. The probability of success for part repair attempts also depends on the part and the site.

The purpose of the inventory controller is to generate, at each time instant t_n, decisions as to the number of parts of each part type that each site should order from its associated supplier site, and (in the case of the manufacturing sites) the number of parts of each part type to start manufacturing so as to meet some performance objective. We consider two possible performance objectives. The first performance objective that we consider is, roughly stated, the reduction of excess inventory or *slack* in the system. In this case, inventory level set points are tuned online through signals that react to the demand profiles, and the controller attempts to satisfy the demand with the lowest possible on-site inventory levels. The second performance objective that we consider is based on a performance index specified in terms of the parts available at the end-nodes. The aircraft supply chain examined in Section IV features a physically meaningful performance index of this kind, the *mission capability* that is defined in terms of a set of requisite parts for a plane to be deemed mission capable. The performance objectives described earlier are characterized more precisely in Section III, and inventory control strategies to meet the performance objectives are developed. It is preferable in the design of the inventory controllers that the amount of information exchange required between sites for the functioning of the controller should be minimal to yield a fully or partially decentralized scheme. In Section III, it is seen that the aforementioned first performance objective can be attained in a fully decentralized framework, while the second objective

[3]Hence, $\mathcal{S}$ should be a function of $\Gamma_{(i,j)}$, p_k, and t. However, to avoid notational complexity, we leave the time dependence implicit in $\mathcal{S}$.

requires information exchange between successive layers in the supply chain.

III. Control Strategies

In this section, we develop inventory control strategies based on the model of supply chains with repairs developed in Section II. First, we formulate a mathematical description appropriate for control design of the model developed in Section II. The following signals are introduced for each site $\Gamma_{(i,j)}, i = 1, \ldots, L-1, j = 1, \ldots, N_i$ at each time instant t_n and for each part type p_k.

1) $r_{\rm cs}(\Gamma_{(i,j)}; p_k; t_n)$: number of repair attempts for parts of type p_k completed successfully at time t_n at site $\Gamma_{(i,j)}$.
2) $r_{\rm cu}(\Gamma_{(i,j)}; p_k; t_n)$: number of repair attempts for parts of type p_k completed unsuccessfully at time t_n at site $\Gamma_{(i,j)}$.
3) $d(\Gamma_{(i,j)}; p_k; t_n)$: number of parts of type p_k received from downstream (i.e., from a site in $\{\Gamma_{(i+1,r)} | r \in \mathcal{S}^{-1}(\Gamma_{(i,j)}; p_k)\}$) at time t_n at site $\Gamma_{(i,j)}$.
4) $u(\Gamma_{(i,j)}; p_k; t_n)$: number of parts of type p_k received from upstream (i.e., from $\mathcal{S}(\Gamma_{(i,j)}; p_k)$) at time t_n at site $\Gamma_{(i,j)}$.
5) $n_d(\Gamma_{(i,j)}; p_k; t_n)$: number of new orders for part type p_k received from downstream at time t_n at site $\Gamma_{(i,j)}$.
6) $n_u(\Gamma_{(i,j)}; p_k; t_n)$: number of new orders for part type p_k placed to upstream at time t_n from site $\Gamma_{(i,j)}$.
7) $s(\Gamma_{(i,j)}; p_k; t_n)$: number of parts of type p_k sent downstream at time t_n from site $\Gamma_{(i,j)}$.

In Section II, we introduced the notation $\Gamma^{p_k}_{(i,j)}(t_n)$ for the number of (working) parts of type p_k in the on-site inventory at site $\Gamma_{(i,j)}$. Also, let $\Gamma^{Rp_k}_{(i,j)}(t_n)$ be the number of parts of type p_k under repair at site $\Gamma_{(i,j)}$ at time t_n. Let $\Gamma^{Up_k}_{(i,j)}(t_n)$ be the number of parts of type p_k expected from upstream at site $\Gamma_{(i,j)}$ at time t_n. Note that since we use a pull architecture for the supply chain, $\Gamma^{Up_k}_{(i,j)}(t_n)$ is a function of the number of parts sent upstream for repair and the number of new orders placed to upstream till the current time. Let $\Gamma^{Op_k}_{(i,j)}(t_n)$ be the number of outstanding orders of type p_k at site $\Gamma_{(i,j)}$ at time t_n, i.e., the number of parts of type p_k that downstream sites are waiting for from site $\Gamma_{(i,j)}$. Let $\Gamma^{Np_k}_{(i,j)}(t_n) \triangleq (\Gamma^{p_k}_{(i,j)}(t_n) + \Gamma^{Rp_k}_{(i,j)}(t_n) + \Gamma^{Up_k}_{(i,j)}(t_n))$ denote the net inventory of parts of type p_k at site $\Gamma_{(i,j)}$ at time t_n. The net inventory includes the parts in the on-site inventory, the parts currently under repair on-site, and the parts expected from upstream. Let $\Gamma^{Pp_k}_{(i,j)}(t_n) \triangleq (\Gamma^{Np_k}_{(i,j)}(t_n) - \Gamma^{Op_k}_{(i,j)}(t_n)) = (\Gamma^{p_k}_{(i,j)}(t_n) + \Gamma^{Rp_k}_{(i,j)}(t_n) + \Gamma^{Up_k}_{(i,j)}(t_n) - \Gamma^{Op_k}_{(i,j)}(t_n))$ denote the *inventory position* for part type p_k at site $\Gamma_{(i,j)}$ at time t_n, i.e., the difference between the net inventory and the outstanding orders. The inventory dynamics at each site $\Gamma_{(i,j)}, i = 1, \ldots, L-1, j = 1, \ldots, N_i$, can be expressed through the following relations:

$$\begin{aligned}\Gamma^{p_k}_{(i,j)}(t_{n+1}) = \Gamma^{p_k}_{(i,j)}(t_n) + r_{cs}(\Gamma_{(i,j)}; p_k; t_n) \\ + u(\Gamma_{(i,j)}; p_k; t_n) - s(\Gamma_{(i,j)}; p_k; t_n)\end{aligned} \tag{4}$$

$$\begin{aligned}\Gamma^{Rp_k}_{(i,j)}(t_{n+1}) = \Gamma^{Rp_k}_{(i,j)}(t_n) - r_{cs}(\Gamma_{(i,j)}; p_k; t_n) \\ - r_{cu}(\Gamma_{(i,j)}; p_k; t_n) + d(\Gamma_{(i,j)}; p_k; t_n)\end{aligned} \tag{5}$$

$$\begin{aligned}\Gamma^{Up_k}_{(i,j)}(t_{n+1}) = \Gamma^{Up_k}_{(i,j)}(t_n) + r_{cu}(\Gamma_{(i,j)}; p_k; t_n) \\ + n_u(\Gamma_{(i,j)}; p_k; t_n) - u(\Gamma_{(i,j)}; p_k; t_n)\end{aligned} \tag{6}$$

$$\begin{aligned}\Gamma^{Op_k}_{(i,j)}(t_{n+1}) = \Gamma^{Op_k}_{(i,j)}(t_n) + d(\Gamma_{(i,j)}; p_k; t_n) \\ + n_d(\Gamma_{(i,j)}; p_k; t_n) - s(\Gamma_{(i,j)}; p_k; t_n)\end{aligned} \tag{7}$$

$$\begin{aligned}\Gamma^{Np_k}_{(i,j)}(t_{n+1}) = \Gamma^{Np_k}_{(i,j)}(t_n) + d(\Gamma_{(i,j)}; p_k; t_n) \\ + n_u(\Gamma_{(i,j)}; p_k; t_n) - s(\Gamma_{(i,j)}; p_k; t_n)\end{aligned} \tag{8}$$

$$\begin{aligned}\Gamma^{Pp_k}_{(i,j)}(t_{n+1}) = \Gamma^{Pp_k}_{(i,j)}(t_n) + n_u(\Gamma_{(i,j)}; p_k; t_n) \\ - n_d(\Gamma_{(i,j)}; p_k; t_n).\end{aligned} \tag{9}$$

The dynamics of a manufacturing site $\Gamma_{(0,j)}$ can be obtained similarly. Let $m(\Gamma_{(0,j)}; p_k; t_n)$ and $m_c(\Gamma_{(0,j)}; p_k; t_n)$ be the numbers of part manufactures of type p_k initiated and completed, respectively, at time t_n at site $\Gamma_{(0,j)}$. Let $\Gamma^{Mp_k}_{(0,j)}(t_n)$ be the number of parts of type p_k under manufacture at site $\Gamma_{(0,j)}$ at time t_n. With the rest of the notations defined analogously to that expressed earlier, the inventory dynamics at each site $\Gamma_{(0,j)}, j = 1, \ldots, N_0$ can be written as

$$\begin{aligned}\Gamma^{p_k}_{(0,j)}(t_{n+1}) = \Gamma^{p_k}_{(0,j)}(t_n) + r_{cs}(\Gamma_{(0,j)}; p_k; t_n) \\ + m_c(\Gamma_{(0,j)}; p_k; t_n) - s(\Gamma_{(0,j)}; p_k; t_n)\end{aligned} \tag{10}$$

$$\begin{aligned}\Gamma^{Rp_k}_{(0,j)}(t_{n+1}) = \Gamma^{Rp_k}_{(0,j)}(t_n) - r_{cs}(\Gamma_{(0,j)}; p_k; t_n) \\ - r_{cu}(\Gamma_{(0,j)}; p_k; t_n) + d(\Gamma_{(0,j)}; p_k; t_n)\end{aligned} \tag{11}$$

$$\begin{aligned}\Gamma^{Mp_k}_{(0,j)}(t_{n+1}) = \Gamma^{Mp_k}_{(0,j)}(t_n) + m(\Gamma_{(0,j)}; p_k; t_n) \\ - m_c(\Gamma_{(0,j)}; p_k; t_n)\end{aligned} \tag{12}$$

$$\begin{aligned}\Gamma^{Op_k}_{(0,j)}(t_{n+1}) = \Gamma^{Op_k}_{(0,j)}(t_n) + d(\Gamma_{(0,j)}; p_k; t_n) \\ + n_d(\Gamma_{(0,j)}; p_k; t_n) - s(\Gamma_{(0,j)}; p_k; t_n)\end{aligned} \tag{13}$$

$$\begin{aligned}\Gamma^{Np_k}_{(0,j)}(t_{n+1}) = \Gamma^{Np_k}_{(0,j)}(t_n) + d(\Gamma_{(0,j)}; p_k; t_n) \\ + m(\Gamma_{(0,j)}; p_k; t_n) - s(\Gamma_{(0,j)}; p_k; t_n) \\ - r_{cu}(\Gamma_{(0,j)}; p_k; t_n)\end{aligned} \tag{14}$$

$$\begin{aligned}\Gamma^{Pp_k}_{(0,j)}(t_{n+1}) = \Gamma^{Pp_k}_{(0,j)}(t_n) + m(\Gamma_{(0,j)}; p_k; t_n) \\ - n_d(\Gamma_{(0,j)}; p_k; t_n) - r_{cu}(\Gamma_{(0,j)}; p_k; t_n).\end{aligned} \tag{15}$$

To derive the inventory dynamics at the end-nodes, the number of parts of type p_k that fail at the site $\Gamma_{(L,j)}(t_n)$ is denoted by $f(\Gamma_{(L,j)}; p_k; t_n)$. With $u(\Gamma_{(L,j)}; p_k; t_n)$, $n_u(\Gamma_{(L,j)}; p_k; t_n)$, $\Gamma^{p_k}_{(L,j)}(t_n)$, $\Gamma^{Up_k}_{(L,j)}(t_n)$, and $\Gamma^{Np_k}_{(L,j)}(t_n)$ defined analogously to that expressed earlier, the inventory dynamics of each site $\Gamma_{(L,j)}, j = 1, \ldots, N_L$ can be expressed as

$$\begin{aligned}\Gamma^{p_k}_{(L,j)}(t_{n+1}) = \Gamma^{p_k}_{(L,j)}(t_n) - f(\Gamma_{(L,j)}; p_k; t_n) \\ + u(\Gamma_{(L,j)}; p_k; t_n)\end{aligned} \tag{16}$$

$$\begin{aligned}\Gamma^{Up_k}_{(L,j)}(t_{n+1}) = \Gamma^{Up_k}_{(L,j)}(t_n) + f(\Gamma_{(L,j)}; p_k; t_n) \\ + n_u(\Gamma_{(L,j)}; p_k; t_n) - u(\Gamma_{(L,j)}; p_k; t_n)\end{aligned} \tag{17}$$

$$\Gamma^{Np_k}_{(L,j)}(t_{n+1}) = \Gamma^{Np_k}_{(L,j)}(t_n) + n_u(\Gamma_{(L,j)}; p_k; t_n). \tag{18}$$

Remark 1: Note that in the considered supply chain model, the part requirements at the end-nodes are that each of the end-nodes $\Gamma_{(L,j)}$ requires $n^{p_k}_{(L,j)}$ items of part type p_k to be considered functional and end-nodes cannot store excess inventory. Assuming, without loss of generality, that at the initialization time t_0, each end-node has the required number of items (which could be working or failed) of each part type, it follows that $n_u(\Gamma_{(L,j)}; p_k; t_n) = 0$ for all $j \in \{1, \ldots, N_L\}$, all $k \in \{1, \ldots, P\}$, and all t_n. If, on the other hand, an end-node is missing some parts at initialization time, then $n_u(\Gamma_{(L,j)}; p_k; t_0)$ is picked to be the difference between $n^{p_k}_{(L,j)}$ and the number of items (working or failed) of part type p_k at end-node $\Gamma_{(L,j)}$ at initialization time, and $n_u(\Gamma_{(L,j)}; p_k; t_n) = 0$ for all $j \in \{1, \ldots, N_L\}$, all $k \in \{1, \ldots, P\}$, and all $t_n > t_0$. The proposed adaptive inventory control algorithm can be extended to allow time-varying end-node part requirements as alluded to earlier by setting $n_u(\Gamma_{(L,j)}; p_k; t_n) = n^{p_k}_{(L,j)}(t_n) - n^{p_k}_{(L,j)}(t_{n-1})$ in the case of an increase in part requirements, i.e., if $n^{p_k}_{(L,j)}(t_n) - n^{p_k}_{(L,j)}(t_{n-1}) > 0$. In the case of a decrease in part requirements, i.e., if $n^{p_k}_{(L,j)}(t_n) - n^{p_k}_{(L,j)}(t_{n-1}) < 0$, then $n_u(\Gamma_{(L,j)}; p_k; t_n)$ is retained to be zero; however, the excess parts are propagated upstream in the absence of a facility for on-site excess inventory storage at end-nodes.

The inventory dynamics of the sites in the supply chain are coupled through the variables d, u, n_d, n_u, and s. For instance, if the times taken for a part to move from a site $\Gamma_{(i,j)}$ to its corresponding supplier site $\mathcal{S}(\Gamma_{(i,j)}; p_k)$ are deterministic constants and denoted by $t^{p_k}_{(i,j)}$, then

$$d(\Gamma_{(i,j)}; p_k; t_n) = \sum_{\chi \in \mathcal{S}^{-1}(\Gamma_{(i,j)}; p_k)} r_{cu}\left(\Gamma_{(i+1,\chi)}; p_k; t_n - t^{p_k}_{(i+1,\chi)}\right) \tag{19}$$

for $i = 0, \ldots, L-2, j = 1, \ldots, N_i, k = 1, \ldots, P$, and

$$d(\Gamma_{(L-1,j)}; p_k; t_n) = \sum_{\chi \in \mathcal{S}^{-1}(\Gamma_{(L-1,j)}; p_k)} f\left(\Gamma_{(L,\chi)}; p_k; t_n - t^{p_k}_{(L,\chi)}\right) \tag{20}$$

for $j = 1, \ldots, N_{L-1}, k = 1, \ldots, P$. Also, if the amount of time to repair a part of type p_k at site $\Gamma_{(i,j)}$ is a deterministic constant and denoted by $t^{Rp_k}_{(i,j)}$, then the following relation holds:

$$d\left(\Gamma_{(i,j)}; p_k; t_n - t^{Rp_k}_{(i,j)}\right) = r_{\text{cs}}(\Gamma_{(i,j)}; p_k; t_n) + r_{\text{cu}}(\Gamma_{(i,j)}; p_k; t_n). \tag{21}$$

The decomposition of d into r_{cs} and r_{cu} is governed by the repair success probabilities that are defined for each site and part. In general, the transportation times between sites and the repair times are random variables so that the right-hand sides of (19)–(21) involve not certain fixed time delays, but a set of random time delays. The variables n_d and n_u are coupled through the equation

$$n_d(\Gamma_{(i,j)}; p_k; t_n) = \sum_{\chi \in \mathcal{S}^{-1}(\Gamma_{(i,j)}; p_k)} n_u\left(\Gamma_{(i+1,\chi)}; p_k; t_n - t^{p_k}_{c(i+1,\chi)}\right) \tag{22}$$

where $t^{p_k}_{c(i,j)}$ is the communication delay between the site $\Gamma_{(i,j)}$ and its supplier site $\mathcal{S}(\Gamma_{(i,j)}; p_k)$ when a new order is placed for an item of part type p_k. Note that $t^{p_k}_{c(i,j)}$ is purely the communication delay for the site $\mathcal{S}(\Gamma_{(i,j)}; p_k)$ to be notified that the site $\Gamma_{(i,j)}$ has placed an order and does not include the lead time to actually procure the part. If such communication is fast enough to be accomplished within one sampling time step (as is usually the case), then the quantities $t^{p_k}_{c(i+1,\chi)}$ in (22) can be taken to be zero.

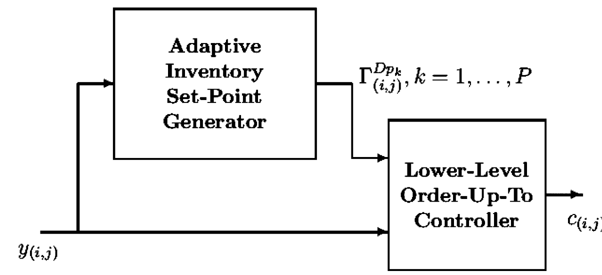

Fig. 2. Hierarchical two-level inventory controller.

The inventory dynamics of the entire supply chain can, at least conceptually, be obtained by combining together the inventory dynamics of each site and by formulating all the coupling signals through the appropriate probability distributions. However, this process is computationally infeasible for any but the simplest supply chains. Instead, we follow here an *agent-based* approach, wherein the dynamics of each site are considered separately and the dynamics of the entire supply chain is implicitly captured through the behavior of each site, the transportation network, and the repair and manufacture processes. This simplifies both the controller design and analysis, and also the computer simulation (see Section V) where the agent-based approach maps naturally to an object-oriented framework.

The control signal that is to be generated by the inventory controller at a site $\Gamma_{(i,j)}$ is $c_{(i,j)} \triangleq \{n_u(\Gamma_{(i,j)}; p_k; t_n) | k = 1, \ldots, P\}$ for $i = 1, \ldots, L-1, j = 1, \ldots, N_i$, and $c_{(i,j)} \triangleq \{m(\Gamma_{(i,j)}; p_k; t_n) | k = 1, \ldots, P\}$ for $i = 0, j = 1, \ldots, N_0$, i.e., the inventory controller is required to make decisions on new orders and manufactures. We propose a two-level hierarchical controller structure with the higher level controller being an adaptive inventory set point generator (i.e., an algorithm that determines the *desired stock level* adaptively at runtime) and the lower level controller following an order-up-to policy. This is in contrast to optimization-based schemes, wherein the inventory set points are fixed through offline optimization. The structure of the proposed controller strategy is illustrated in Fig. 2, where the input to the inventory controller is denoted by $y_{(i,j)}$. The proposed controller is of a simple form with low-computational requirements, thus making application to large-scale supply chains feasible. Furthermore, the adaptive strategy allows the supply chain to react rapidly to changes in the topology of the supply chain network and supplier relationships. We propose two different adaptive higher level controllers, one fully decentralized and the other partially decentralized. In the fully decentralized case, $y_{(i,j)}$ is composed of local measurements of $\Gamma^{Np_k}_{(i,j)}$ and $\Gamma^{Op_k}_{(i,j)}$. In the partially decentralized case, $y_{(i,j)}$ also incorporates *part deficit* signals received from the sites for which $\Gamma_{(i,j)}$ acts as a supplier. The dynamic performance of the proposed controllers is evaluated through simulation studies in Section V.

Let the inventory set point (i.e., the desired inventory level) for part type p_k at site $\Gamma_{(i,j)}$ at time t_n be denoted by $\Gamma^{Dp_k}_{(i,j)}(t_n)$. The lower level controller works to regulate the net inventory $\Gamma^{Np_k}_{(i,j)}(t_n)$ to the inventory set point $\Gamma^{Dp_k}_{(i,j)}(t_n)$, while the adaptive higher level controller performs online tuning of the inventory set points $\Gamma^{Dp_k}_{(i,j)}(t_n)$. The function of the $\Gamma^{Dp_k}_{(i,j)}$ signal is to drive the inventory level $\Gamma^{p_k}_{(i,j)}$ for part type p_k at site $\Gamma_{(i,j)}$ to an appropriate level that captures the supply chain and part failure characteristics. The lower level controller that directly assigns $n_u(\Gamma_{(i,j)}; p_k; t_n), i = 1,\ldots,L-1, j = 1,\ldots,N_i, k = 1,\ldots,P$ and $m(\Gamma_{(0,j)}; p_k; t_n), j = 1,\ldots,N_0, k = 1,\ldots,P$ is given by

$$n_u(\Gamma_{(i,j)}; p_k; t_n) = \begin{cases} 0 \text{ if } \Gamma^{Np_k}_{(i,j)}(t_n) \geq \Gamma^{Dp_k}_{(i,j)}(t_n) \\ \Gamma^{Dp_k}_{(i,j)}(t_n) - \Gamma^{Np_k}_{(i,j)}(t_n) \text{ otherwise} \end{cases} \tag{23}$$

for $i = 1,\ldots,L-1, j = 1,\ldots,N_i, k = 1,\ldots,P$ and

$$m(\Gamma_{(0,j)}; p_k; t_n) = \begin{cases} 0 \text{ if } \Gamma^{Np_k}_{(0,j)}(t_n) \geq \Gamma^{Dp_k}_{(0,j)}(t_n) \\ \Gamma^{Dp_k}_{(0,j)}(t_n) - \Gamma^{Np_k}_{(0,j)}(t_n) \text{ otherwise} \end{cases} \tag{24}$$

for $j = 1,\ldots,N_0, k = 1,\ldots,P$.

We first consider a fully decentralized candidate for the adaptive higher level controller given by

$$\begin{aligned} \Gamma^{Dp_k}_{(i,j)}(t_n) &= \max\left\{0, \tilde{\Gamma}^{Dp_k}_{(i,j)}(t_n)\right\} \\ \tilde{\Gamma}^{Dp_k}_{(i,j)}(t_n) &= C^{p_k}_{(i,j)P}\tilde{\Gamma}^{Op_k}_{(i,j)}(t_n) \\ &\quad + C^{p_k}_{(i,j)D}\left[\tilde{\Gamma}^{Op_k}_{(i,j)}(t_n) - \tilde{\Gamma}^{Op_k}_{(i,j)}(t_{n-1})\right] \end{aligned} \tag{25}$$

where $\tilde{\Gamma}^{Op_k}_{(i,j)}(t_n)$ is a low-pass filtered version[4] of $\Gamma^{Op_k}_{(i,j)}(t_n)$. $C^{p_k}_{(i,j)P}$ and $C^{p_k}_{(i,j)D}$ are nonnegative constants and form the controller-gain parameters. The controller (25) is essentially a proportional-derivative (PD) controller (i.e., a control law including two terms, the first proportional to the error signal, which in this case is $\Gamma^{Op_k}_{(i,j)}$, and the second term proportional to the rate of change of the error signal) based on $\Gamma^{Op_k}_{(i,j)}(t_n)$. The use of a low-pass filtered version of $\Gamma^{Op_k}_{(i,j)}(t_n)$ in (25) rather than $\Gamma^{Op_k}_{(i,j)}(t_n)$ itself reduces the sensitivity to stochastically induced local spikes. The bandwidth of the low-pass filter should be picked, based on estimates of the time constants of the system that can be inferred from mean time before failure of each part, transportation delays, and repair and manufacture times.

The controller (25) at each node and the overall inventory dynamics of the supply chain form a closed-loop system. To prove the well behavedness or *stability* (i.e., boundedness of all signals at each node) of this closed-loop system, consider the following relations that can be derived from the part requirements specified at the end-nodes

$$\overline{n}(L,j,p_k,t_n) \geq d(\Gamma_{(L-1,j)}; p_k; t_n) + n_d(\Gamma_{(L-1,j)}; p_k; t_n) \tag{26}$$

$$\overline{n}(L,j,p_k,t_n) \geq \Gamma^{Op_k}_{(L-1,j)}(t_n) \tag{27}$$

where $\overline{n}(L,j,p_k,t_n) \triangleq \sum_{\chi \in \mathcal{S}^{-1}(\Gamma_{(L-1,j)}; p_k; t_n)} n^{p_k}_{(L,\chi)}$. Since $\tilde{\Gamma}^{Op_k}_{(L-1,j)}(t_n)$ is a low-pass filtered version of $\Gamma^{Op_k}_{(L-1,j)}(t_n)$, it follows from (27) that

$$\tilde{\Gamma}^{Op_k}_{(L-1,j)}(t_n) \leq \overline{n}(L,j,p_k,t_n) \tag{28}$$

$$\tilde{\Gamma}^{Op_k}_{(L-1,j)}(t_n) - \tilde{\Gamma}^{Op_k}_{(L-1,j)}(t_{n-1}) \leq 2\overline{n}(L,j,p_k,t_n). \tag{29}$$

The inequalities (27)–(29) and the control law (25) yield the inequality

$$\Gamma^{Dp_k}_{(L-1,j)}(t_n) \leq \left(C^{p_k}_{(L-1,j)P} + 2C^{p_k}_{(L-1,j)D}\right)\overline{n}(L,j,p_k,t_n). \tag{30}$$

This provides a uniform upper bound on $\Gamma^{Dp_k}_{(L-1,j)}(t_n)$. Bounds on $\Gamma^{Dp_k}_{(i,j)}(t_n), i = L-2,\ldots,0$ can be obtained using induction via inequalities analogous to (26) and (27). It can be seen that boundedness of the signals $\Gamma^{Dp_k}_{(i,j)}(t_n), i = 0,\ldots,L-1$ implies boundedness of all signals including the stock levels and upstream part orders at each node, thus implying stability of the closed-loop system formed by the overall inventory dynamics of the supply chain and the designed controller.

The higher level controller (25) is completely decentralized and does not require any information transfer (in addition to the information transfer required by the supply chain itself, i.e., the part transfer and the order placement links) between sites in the supply chain. The downstream demand profiles are inferred purely through the local measurements of broken parts arriving and new orders being placed. If information transfer links between sites and the associated supplier sites can be exploited in the controller, then the performance can be further improved by passing downstream demand information directly to the controller at the supplier site. Furthermore, a performance index defined at the end-nodes can be taken into account in the controller decisions at the upstream sites. Consider a performance index of the form $P_{(L,j)}(\Gamma^{p_1}_{(L,j)},\ldots,\Gamma^{p_P}_{(L,j)})$ defined at each end-node $\Gamma_{(L,j)}$. The performance index is decomposed into part deficit signals $P^{p_k}_{(L,j)}(t_n)$ defined for each part type p_k at each end-node $\Gamma_{(L,j)}$. The part deficit signals indicate the shortage of each part type at each end-node. If, as discussed in Section II, the performance requirement at the end-nodes is that each end-node $\Gamma_{(L,j)}$ requires $n^{p_k}_{(L,j)}$ items of part type p_k to be functional, then the part deficit signals at the end-nodes are defined as $P^{p_k}_{(L,j)}(t_n) = n^{p_k}_{(L,j)} - \Gamma^{p_k}_{(L,j)}(t_n), k = 1,\ldots,P$. The adaptive higher level controllers at the upstream sites are

[4]A low-pass filter [19] attenuates high-frequencies contained in the input signal, which is $\Gamma^{Op_k}_{(i,j)}(t_n)$ in this case, but passes through low frequencies, thus smoothing out short-term variations (noise). A simple low-pass filter is given by the recursive relation $\tilde{\Gamma}^{Op_k}_{(i,j)}(t_n) = a\Gamma^{Op_k}_{(i,j)}(t_n) + (1-a)\tilde{\Gamma}^{Op_k}_{(i,j)}(t_{n-1})$ with the initial condition $\tilde{\Gamma}^{Op_k}_{(i,j)}(t_0) = \Gamma^{Op_k}_{(i,j)}(t_0)$ at the initialization time t_0. a can be chosen to be any constant in the interval $(0,1)$. A smaller value (i.e., closer to 0) of a results in a smoother output signal $\tilde{\Gamma}^{Op_k}_{(i,j)}(t_n)$, while a larger value (i.e., closer to 1) of a results in passing through more short-term variations into the output signal. Denoting the sampling period $(t_n - t_{n-1})$ by T_s, the *bandwidth* of the low-pass filter is defined to be $a/[(1-a)T_s]$.

defined inductively as

$$
\begin{aligned}
P^{p_k}_{(i,j)}(t_n) &= \overline{P}^{p_k}_{(i,j)}(t_n) \\
\Gamma^{Dp_k}_{(i,j)}(t_n) &= \max\left\{0, \tilde{\Gamma}^{Dp_k}_{(i,j)}(t_n)\right\} \\
\tilde{\Gamma}^{Dp_k}_{(i,j)}(t_n) &= C^{p_k}_{(i,j)P}\tilde{\Gamma}^{Op_k}_{(i,j)}(t_n) + C^{p_k}_{(i,j)D}\left[\tilde{\Gamma}^{Op_k}_{(i,j)}(t_n)\right. \\
&\quad \left. - \tilde{\Gamma}^{Op_k}_{(i,j)}(t_{n-1})\right] + P^{p_k}_{(i,j)} f_E\left(\Gamma^{Pp_k}_{(i,j)}(t_n)\right) \qquad (31)
\end{aligned}
$$

if $\Gamma^{Pp_k}_{(i,j)}(t_n) \geq 0$ and

$$
\begin{aligned}
P^{p_k}_{(i,j)}(t_n) &= f_C\left(\left|\Gamma^{Pp_k}_{(i,j)}(t_n)\right|\right)\overline{P}^{p_k}_{(i,j)}(t_n) \\
\Gamma^{Dp_k}_{(i,j)}(t_n) &= \max\left\{0, \tilde{\Gamma}^{Dp_k}_{(i,j)}(t_n)\right\} \\
\tilde{\Gamma}^{Dp_k}_{(i,j)}(t_n) &= C^{p_k}_{(i,j)P}\tilde{\Gamma}^{Op_k}_{(i,j)}(t_n) + C^{p_k}_{(i,j)D}\left[\tilde{\Gamma}^{Op_k}_{(i,j)}(t_n)\right. \\
&\quad \left. - \tilde{\Gamma}^{Op_k}_{(i,j)}(t_{n-1})\right] + P^{p_k}_{(i,j)} f_D\left(\left|\Gamma^{Pp_k}_{(i,j)}(t_n)\right|\right) \qquad (32)
\end{aligned}
$$

if $\Gamma^{Pp_k}_{(i,j)}(t_n) < 0$ where

$$\overline{P}^{p_k}_{(i,j)}(t_n) = \sum_{\chi \in \mathcal{S}^{-1}(\Gamma_{(i,j)}; p_k; t_n)} P^{p_k}_{(i+1,\chi)}. \qquad (33)$$

The functions f_C and f_D are picked to be increasing functions, while f_E is picked to be a decreasing function. The controller given in (31) and (32) is essentially based on translating the part deficit signals of the downstream sites into on-site generated part deficit signals to be further passed on to upstream sites. The part deficit signals provide an estimate of the degree of shortage of each part type at each node and enable a feedforward action in the controller, thus providing faster response to changes in the supply chain. While the computation of $\tilde{\Gamma}^{Dp_k}_{(i,j)}(t_n)$ in the fully decentralized controller (25) does not require any information transfer between sites in the supply chain, the computations of $\tilde{\Gamma}^{Dp_k}_{(i,j)}(t_n)$ in (31) and (32) utilize the signals $P^{p_k}_{(i+1,\chi)}$ communicated from the downstream sites for which the site $\Gamma_{(i,j)}$ acts as a supplier. The implementation of the controller requires information transfer between each site and the associated supplier sites. Recall that the inventory position $\Gamma^{Pp_k}_{(i,j)}$ is defined as the difference between the net inventory and the outstanding orders. If $\Gamma^{Pp_k}_{(i,j)}(t_n)$ is positive, then there is a potential of future inventory surplus when the parts under repair and the parts expected from upstream enter the working inventory with the surplus being higher, if $\Gamma^{Pp_k}_{(i,j)}(t_n)$ is higher. Hence, the feedforward component $P^{p_k}_{(i,j)}$ in (31) that reacts to part deficit signals from downstream sites is attenuated by the use of the term $f_E(\Gamma^{Pp_k}_{(i,j)}(t_n))$, with f_E picked to be a decreasing function to avoid inventory surplus. In the case that $\Gamma^{Pp_k}_{(i,j)}(t_n)$ is negative, there is a potential of future inventory deficit to avoid which the terms $f_C(\Gamma^{Pp_k}_{(i,j)}(t_n))$ and $f_D(\Gamma^{Pp_k}_{(i,j)}(t_n))$ are used in (32) to amplify the effect of the feedforward component which reacts to part deficit signals from downstream sites. The stability analysis of this partially decentralized controller can be carried out along similar lines to the aforementioned fully decentralized controller.

Remark 2: The performance can be further improved at the expense of increased computation and communication requirements by considering possibly overlapped geographical conglomerations of sites that behave as larger metasites with a cooperative inventory level adaptation. For instance, a site and a set of its supplier sites can be grouped into a larger metasite with the inventory set points for the metasite being controlled using either of the controllers developed earlier. This provides a possibility of reducing inventories while also reducing transients in the closed-loop system. The mathematical foundation for such groupings of sites is provided by the overlapping decomposition theory [20]–[22].

Remark 3: For simplicity and brevity in this paper, we have considered the case in which each site has a single supplier for each part type. However, the proposed control algorithms and the simulation framework can be easily extended to allow the multiple supplier scenario, wherein a site is allowed to order a part type from multiple supplier sites. In this case, if a large order needs to be placed to upstream from a site, the site could split it into multiple orders to more than one of its supplier sites to speed up the delivery. Furthermore, in the case of the partially decentralized controller, the part deficit signals from the upstream supplier sites could be used to detect which among the supplier sites is most likely to supply (or repair) the required part most expeditiously.

IV. Application to an Aircraft Supply Chain

The aircraft supply chain model consists of OEMs, depots, bases, squadrons, and planes

$$\text{OEM} \leftrightarrow \text{Depot} \leftrightarrow \text{Base} \leftrightarrow \text{Squadron} \leftrightarrow \text{Plane}. \qquad (34)$$

The new and repaired parts move from left to right in the supply chain in (34), while the requests for new parts and repair move from right to left. The OEMs, depots, and bases can attempt part repair, while only the OEMs can manufacture new parts. Part inventory stocks are maintained at OEMs, depots, bases, and squadrons. The end-nodes, i.e., the planes, have an associated set of parts, the availability of which determines the *mission capability* (MC).

When a part on a plane fails, the broken part is propagated up the supply chain, i.e., right to left in (34). Each site that is shipped a broken part attempts to repair it. If the repair is successful, the repaired part is returned downstream based on a first-in-first-out queue of part requests maintained at each site. If a repair is unsuccessful, the broken part is shipped further upstream. Typically, sites that are located further upstream have superior technical facilities, and hence, higher probability of successful part repair. If repair attempts for a part are repeatedly unsuccessful, the part eventually propagates up to an OEM, which also attempts repair of the part. If unsuccessful, the OEM condemns the part and builds a new part.

The aircraft supply chain falls into the general class of supply chains described in Section II, and hence, the adaptive inventory control strategies developed in Section III are applicable to the aircraft supply chain. Also, as mentioned in Section III, the transient performance of the overall supply chain can be improved while also reducing inventory set points at the expense of increased computation and communication

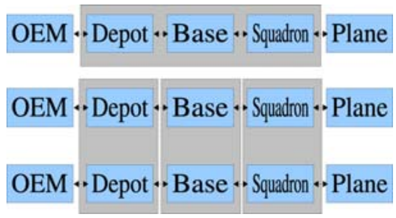

Fig. 3. Grouping sites into metasites.

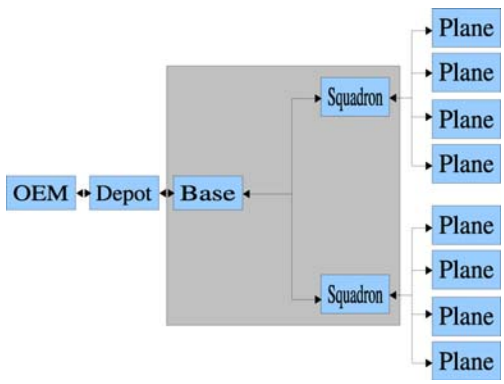

Fig. 4. Aircraft supply chain.

requirements by considering groupings of sites into metasites (see Fig. 3). Usually, the squadrons and the associated base are colocated so that the squadrons and the corresponding base do not need to maintain separate inventories. In such a case, squadrons along with the associated base can be grouped together into a metasite, as shown in Fig. 4.

V. Simulation Results

We have developed a simulation package for the class of supply chains described in Section II using an agent-based framework. The simulation package is written in the powerful object-oriented language Python [23]. The object-oriented programming model [24] through the use of abstraction and the encapsulation of data and functionality in objects with well-defined physical meaning greatly facilitates an agent-based framework and a behavioral description of the sites and the parts. The behavior of a manufacturing site, an intermediate site, an end-node, a part type, and the transportation network are specified in terms of Python classes, and each site and part are created as objects from the associated class. The object-oriented framework provides easy reconfigurability and scalability of the simulation package. This provides a flexible framework with support for arbitrary network topologies with any numbers of sites, parts, and part types. The numbers and locations of sites and parts can be specified at run-time. Part failures, transport delays, and manufacturing and repair times are randomly generated by using probability distributions that are specified at run-time.

We first consider the aircraft supply chain shown in Fig. 4, which consists of one OEM, one depot, one base, two squadrons, and four planes per squadron. As is usually the case, the squadrons and the base are taken to be colocated. Hence, as pointed out in Section IV, the squadrons and the base do not need to maintain separate inventories, and can be grouped together into a metasite, as shown in Fig. 4. For simplicity, the parts requirement for the planes is taken to consist of only two part types p_1 and p_2. Each plane is required to have one part each of types p_1 and p_2 to be considered mission capable. The time-step Δt for the simulation is taken to be 1 day. The times before failure of the part types p_1 and p_2 are taken to be governed by Gaussian distributions with means 10 days and 20 days, respectively for part types p_1 and p_2, and standard deviations 3 days and 4 days, respectively. The transportation times from the base to the depot and from the depot to the OEM are taken to be either 3, 4, or 5 days with each alternative having probability 1/3. The probabilities of a successful repair at the base, depot, and OEM are taken to be 0.75, 0.85, and 0.9, respectively. The time taken for a repair attempt at each of the base, depot, and OEM is taken to be either 1 day or 2 days with each alternative having probability 0.5. The time taken to manufacture a part at the OEM is also taken to be either 1 day or 2 days with each alternative having probability 0.5. The supply chain is initialized with each plane having one each of part types p_1 and p_2, and with each of the base, depot, and OEM having three each of each part type. The simulation results with the fully decentralized controller are illustrated in Fig. 6. The controller parameters $C^{p_k}_{(i,j)P}$ and $C^{p_k}_{(i,j)D}$ are taken to be 5 and 1 for each site. The signals $\tilde{\Gamma}^{Op_k}_{(i,j)}$ are obtained through the low-pass filtering $\tilde{\Gamma}^{Op_k}_{(i,j)}(t_n) = 0.1\Gamma^{Op_k}_{(i,j)}(t_n) + 0.9\tilde{\Gamma}^{Op_k}_{(i,j)}(t_{n-1})$. The average mission capability of the planes (i.e., the average percentage of time that each plane was mission capable with the average taken over all the planes) in the closed-loop supply chain with the fully decentralized controller is obtained to be 98.55%. It can be shown that the controller parameters can be used to trade off the average mission capability against the inventory levels. For instance, increasing $C^{p_k}_{(i,j)P}$ to 10 was found, by simulation, to increase the average mission capability to 99.9% while resulting in maximum inventories of 16 of part type p_1 and 7 of part type p_2, attained at the OEM and the base, respectively.

The simulation results for the more large-scale aircraft supply chain shown in Fig. 5 are illustrated in Fig. 7. The supply chain in Fig. 5 consists of two OEMs, ten depots per OEM, ten bases per depot, ten squadrons per base, and ten planes per squadron amounting to a total of 22 222 sites. As in the aforementioned first simulation example, each plane is required to have one each of two part types p_1 and p_2 for mission capability. The probabilities of successful part repairs and the probability distributions for repair times, manufacture times, and times before failure are taken to be as in the aforementioned first simulation example. The transportation times are also taken to be identical to the earlier case, i.e., it takes a part either 3, 4, or 5 days (with equal probabilities) to move from a base to a depot or from a depot to an OEM. Also, the squadrons are taken to be colocated with the associated base. The controller parameters $C^{p_k}_{(i,j)P}$ and $C^{p_k}_{(i,j)D}$ are chosen to be 3 and 1 for each site. The supply chain is initialized with each plane having one each of part types p_1 and p_2, and with each of the base, depot, and OEM having three each of each part type. Fig. 7 shows the average desired inventories,

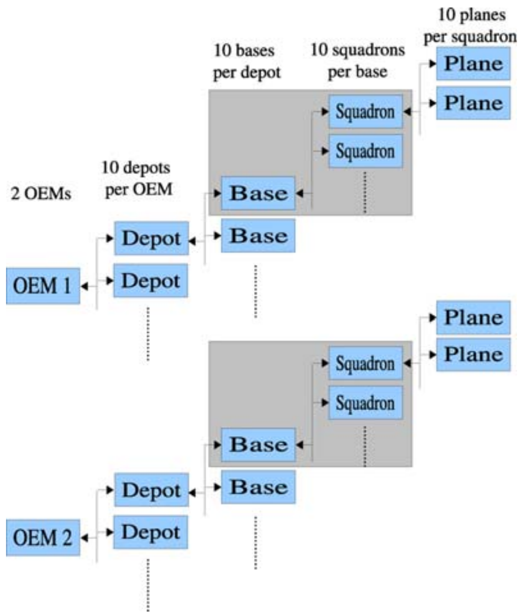

Fig. 5. Large-scale aircraft supply chain.

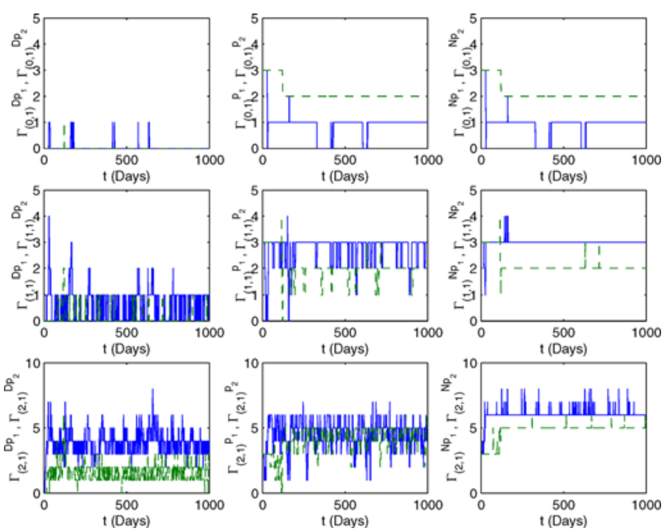

Fig. 6. Simulation results for the aircraft supply chain in Fig. 4. Solid line, p_1 and dashed line, p_2.

on-site inventories, and net inventories with the averages computed over each site type. On a Pentium IV 2.0-GHz desktop computer with 1 GB RAM, the simulation of this large-scale supply chain over a time interval of 1000 days takes around 25 min. Also note that, in implementation, the computations required at each node in the supply chain are minimal, since each node only needs to compute the control law (25) at each time t_n.

It can be shown that the initial transients in the closed-loop supply chain are reduced if the initial inventory levels are increased. The adaptive nature of the inventory controller enables the supply chain to dynamically adapt to changing topologies and supplier relationships. This is demonstrated by introducing a perturbation at time $t = 500$ days at which time the depots associated with OEM2 are reassigned to OEM1, i.e.,

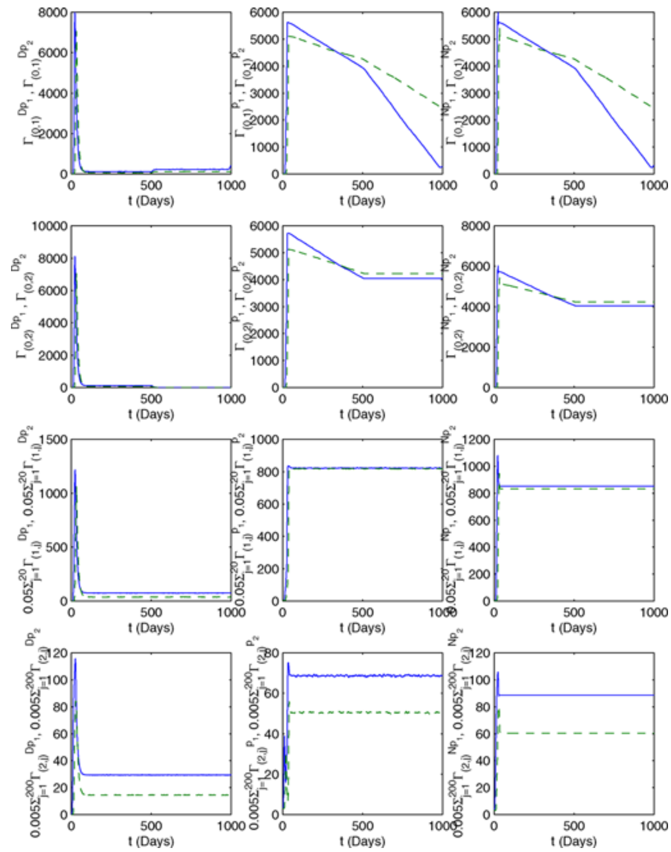

Fig. 7. Simulation results for the aircraft supply chain in Fig. 5. Solid line, p_1 and dashed line, p_2.

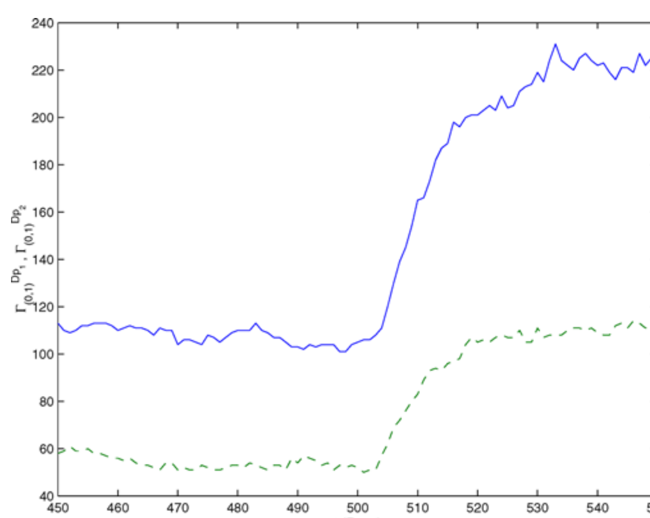

Fig. 8. Adaptation of $\Gamma_{(0,1)}^{Dp_k}$ in response to changes in supplier relationships. Solid line, p_1 and dashed line, p_2.

for $t \geq 500$ days, the supplier for all the depots is OEM1. The adaptation of $\Gamma_{(0,1)}^{Dp_k}$ in response to the increased demand seen by OEM1 is shown in Fig. 8. The average mission capability of the planes in the closed-loop supply chain is obtained to be 98.9%. At steady state, the desired inventory levels for parts p_1 and p_2 at OEM1 are 260 and 115, respectively.

TABLE I
AVERAGE MISSION CAPABILITIES ATTAINED BY ORDER-UP-TO CONTROLLERS USING PRESPECIFIED FIXED INVENTORY SET POINTS

Prespecified desired inventory levels	Average MC
$\Gamma^{Dp_k}_{(0,j)} = 0$ for $j \in \{1,2\}, k \in \{1,2\}$ $\Gamma^{Dp_k}_{(1,j)} = 0$ for $j \in \{1,\ldots,20\}, k \in \{1,2\}$ $\Gamma^{Dp_k}_{(2,j)} = 0$ for $j \in \{1,\ldots,200\}, k \in \{1,2\}$	42.7%
$\Gamma^{Dp_k}_{(0,j)} = 10$ for $j \in \{1,2\}, k \in \{1,2\}$ $\Gamma^{Dp_k}_{(1,j)} = 10$ for $j \in \{1,\ldots,20\}, k \in \{1,2\}$ $\Gamma^{Dp_k}_{(2,j)} = 10$ for $j \in \{1,\ldots,200\}, k \in \{1,2\}$	44.5%
$\Gamma^{Dp_k}_{(0,j)} = 50$ for $j \in \{1,2\}, k \in \{1,2\}$ $\Gamma^{Dp_k}_{(1,j)} = 50$ for $j \in \{1,\ldots,20\}, k \in \{1,2\}$ $\Gamma^{Dp_k}_{(2,j)} = 50$ for $j \in \{1,\ldots,200\}, k \in \{1,2\}$	48%
$\Gamma^{Dp_k}_{(0,j)} = 100$ for $j \in \{1,2\}, k \in \{1,2\}$ $\Gamma^{Dp_k}_{(1,j)} = 100$ for $j \in \{1,\ldots,20\}, k \in \{1,2\}$ $\Gamma^{Dp_k}_{(2,j)} = 100$ for $j \in \{1,\ldots,200\}, k \in \{1,2\}$	64.1%
$\Gamma^{Dp_1}_{(0,j)} = 250$, $\Gamma^{Dp_2}_{(0,j)} = 100$ for $j \in \{1,2\}$ $\Gamma^{Dp_k}_{(1,j)} = 500$ for $j \in \{1,\ldots,20\}, k \in \{1,2\}$ $\Gamma^{Dp_k}_{(2,j)} = 50$ for $j \in \{1,\ldots,200\}, k \in \{1,2\}$	92.6%
$\Gamma^{Dp_1}_{(0,j)} = 260$, $\Gamma^{Dp_2}_{(0,j)} = 115$ for $j \in \{1,2\}$ $\Gamma^{Dp_1}_{(1,j)} = 763$, $\Gamma^{Dp_2}_{(1,j)} = 737$ for $j \in \{1,\ldots,20\}$ $\Gamma^{Dp_1}_{(2,j)} = 65$, $\Gamma^{Dp_2}_{(2,j)} = 46$ for $j \in \{1,\ldots,200\}$	98.4%

The steady-state average inventory levels at the depots (i.e., $0.05\sum_{j=1}^{20}\Gamma^{p_1}_{(1,j)}$ and $0.05\sum_{j=1}^{20}\Gamma^{p_2}_{(1,j)}$) are 763 and 737, respectively, while the steady-state average inventory levels at the bases (i.e.,$0.005\sum_{j=1}^{200}\Gamma^{p_1}_{(2,j)}$ and $0.005\sum_{j=1}^{200}\Gamma^{p_2}_{(2,j)}$) are 65 and 46, respectively.

As a baseline for performance comparison, simulations were also performed with a few different values of prespecified fixed inventory level set points for each site, and with a simple order-up-to controller running at each site. The simulation results are shown in Table I. As could be expected, it is seen that as the prespecified fixed values of the inventory level set points approach the steady-state values of inventory levels, as seen in Fig. 7, the attained average mission capability approaches that of the fully decentralized controller discussed earlier, thus illustrating the efficacy of the proposed adaptive inventory control scheme that computes at runtime appropriate inventory levels based on local measurements, which reflect the overall characteristics of the supply chain including transportation delays, part failure times, manufacturing and repair times, probability of part repair success, etc. While the explicit optimization problem addressing all these effects is not computationally tractable with existing techniques, the adaptive inventory set point generator provides a way to compute appropriate inventory level set points, taking into account the various random effects. It is important to note that the proposed inventory control technique does not require *a priori* availability of data on the supply chain characteristics. However, in this context, it is interesting to note that the proposed inventory control technique in conjunction with the simulation framework can also be used as an efficient offline planning tool to compute inventory level set points if data on the supply chain characteristics are indeed available. The computed inventory levels can then be used, for instance, as initial values at run-time.

To evaluate the performance improvement that can be attained through the use of information transfer links between each site and its associated supplier sites, the supply chain shown in Fig. 5 was simulated with the partially decentralized controller given in (31) and (32) with the controller parameters $C^{p_k}_{(i,j)P}$ and $C^{p_k}_{(i,j)D}$ chosen to be 3 and 1 for each site, and with

$$f_C\left(\Gamma^{Pp_k}_{(i,j)}\right) = f_D\left(\Gamma^{Pp_k}_{(i,j)}\right) = 1 + \left\lfloor \frac{2\Gamma^{Pp_k}_{(i,j)}}{1+\Gamma^{Pp_k}_{(i,j)}} \right\rfloor \tag{35}$$

$$f_E\left(\Gamma^{Pp_k}_{(i,j)}\right) = 1 - \left\lfloor \frac{\Gamma^{Pp_k}_{(i,j)}}{1+\Gamma^{Pp_k}_{(i,j)}} \right\rfloor. \tag{36}$$

The average mission capability with the partially decentralized controller was observed to be 99%, a modest performance improvement over the fully decentralized controller. However, it can be expected that the utility of information transfer links would be more significant when the transportation delays between sites is higher. For instance, if the transportation time between a base and a depot or a depot and an OEM is between 25 and 30 days (with each of the six possibilities having equal probabilities), the average mission capability attained with the fully decentralized controller reduces to 90.5%, while the average mission capability attained with the partially decentralized controller is 95.5%.

VI. CONCLUSION

In this paper, we proposed a new adaptive inventory control strategy wherein the inventory stock set points at each site are tuned via an online adaptation. This yields a technique with low computational requirements that scales well to large supply chain networks. While the controller was designed based on an inherently decentralized control objective, it is seen both from the analysis in Section III and the simulation results in Section V that the developed controllers provide overall performance and efficiency of the supply chain. We conjecture that performance properties of the overall closed-loop supply chain can be proved in an inverse optimality setting [25], [26], and this forms a topic of future research. Other topics for future work include the further relaxation of assumptions and the considering of transportation and storage constraints that induce more coupling between different part types.

REFERENCES

[1] M. C. Bonney, "Trends in inventory management," *Int. J. Product. Econ.*, vol. 35, no. 1–3, pp. 107–114, Jun. 1994.
[2] M. Muller, *Essentials of Inventory Management*. New York: Amer. Manage. Assoc., 2002.
[3] S. Nahmias, *Production and Operations Analysis*. New York: McGraw-Hill/Irwin, 2004.
[4] P. A. Slats, B. Bhola, J. J. M. Evers, and G. Dijkhuizen, "Logistic chain modelling," *Eur. J. Oper. Res.*, vol. 87, no. 1, pp. 1–20, Nov. 1995.
[5] F. Blanchini, R. Pesenti, F. Rinaldi, and W. Ukovich, "Feedback control of production-distribution systems with unknown demand and delays," *IEEE Trans. Robot. Autom.*, vol. 16, no. 3, pp. 313–317, Jun. 2000.

[6] Y. Zhang, P. Luh, K. Narimatsu, T. Moriya, T. Shimada, and L. Fang, "A macrolevel scheduling method using Lagrangian relaxation," *IEEE Trans. Robot. Autom.*, vol. 17, no. 1, pp. 70–79, Feb. 2001.

[7] C. Panayiotou, C. Cassandras, and P. Zhang, "On-line inventory cost minimization for make-to-stock manufacturing systems," in *Proc. Amer. Control Conf.*, Anchorage, AK, May 2002, pp. 4469–4474.

[8] S. Shervais, T. Shannon, and G. Lendaris, "Intelligent supply chain management using adaptive critic learning," *IEEE Trans. Syst., Man, Cybern.*, vol. 33, no. 2, pp. 235–244, Mar. 2003.

[9] T. Nagatani and D. Helbing, "Stability analysis and stabilization strategies for linear supply chains," *Phys. A*, vol. 335, no. 3/4, pp. 644–660, Apr. 2004.

[10] X. Qi, J. F. Bard, and G. Yu, "Supply chain coordination with demand disruptions," *Omega*, vol. 32, no. 4, pp. 301–312, Aug. 2004.

[11] J. Dong, D. Zhang, and A. Nagurney, "A supply chain network equilibrium model with random demands," *Eur. J. Oper. Res.*, vol. 156, no. 1, pp. 194–212, Jul. 2004.

[12] P.-H. Lin, D. S.-H. Wong, S.-S. Jang, S.-S. Shieh, and J.-Z. Chu, "Controller design and reduction of bullwhip for a model supply chain system using z-transform analysis," *J. Process Control*, vol. 14, no. 5, pp. 487–499, Aug. 2004.

[13] M. Dong and F. F. Chen, "Performance modeling and analysis of integrated logistic chains: An analytic framework," *Eur. J. Oper. Res.*, vol. 162, no. 1, pp. 83–98, Apr. 2005.

[14] E. J. Marien, "Reverse logistics as competitive strategy," *Supply Chain Manage. Rev.*, vol. 2, no. 1, pp. 43–52, 1998.

[15] A. Nagurney and F. Toyasaki, "Reverse supply chain management and electronic waste recycling: A multitiered network equilibrium framework for e-cycling," *Transp. Res. E: Logist. Transp. Rev.*, vol. 41, no. 1, pp. 1–28, Jan. 2005.

[16] F. Schultmann, M. Zumkeller, and O. Rentz, "Modeling reverse logistic tasks within closed-loop supply chains: An example from the automotive industry," *Eur. J. Oper. Res.*, vol. 171, no. 3, pp. 1033–1050, Jun. 2006.

[17] A. A. Ghobbar and C. H. Friend, "Evaluation of forecasting methods for intermittent parts demand in the field of aviation: A predictive model," *Comput. Oper. Res.*, vol. 30, no. 14, pp. 2097–2114, Dec. 2003.

[18] E. Esposito and R. Passaro, "Material requirement planning and the supply chain at Alenia Aircraft," *Eur. J. Purch. Supply Manage.*, vol. 3, no. 1, pp. 43–51, Mar. 1997.

[19] B. Porat, *A Course in Digital Signal Processing*. New York: Wiley, 2001.

[20] A. Iftar and F. Khorrami, "A comparison of multiple time-scale analysis and overlapping decompositions," *IEEE Trans. Syst., Man, Cybern.*, vol. 19, no. 5, pp. 1296–1300, Sep.–Oct. 1989.

[21] A. Iftar, "Overlapping decomposition of traffic networks," in *Proc. IEEE Conf. Decision Control*, Tucson, AZ, Dec.1992, pp. 1016–1017.

[22] A. Iftar and U. Ozguner, "Overlapping decompositions, expansions, contractions, and stability of hybrid systems," *IEEE Trans. Autom. Control*, vol. 43, no. 8, pp. 1040–1055, Aug. 1998.

[23] Python programming language [Online]. Available: http://www.python.org

[24] T. Budd, *An Introduction to Object-Oriented Programming*. New York: Addison Wesley, 2001.

[25] H. Deng and M. Krstic, "Stochastic nonlinear Lyapunov stabilization and inverse optimality," in *Proc. IEEE Conf. Decision Control*, San Diego, CA, Dec.1997, pp. 549–554.

[26] K. Ezal, P. Kokotovic, A. Teel, and T. Basar, "Disturbance attenuating output-feedback control of nonlinear systems with local optimality," in *Proc. Amer. Control Conf.*, San Diego, CA, Jun.1999, pp. 2578–2583.

Prashanth Krishnamurthy (S'00–M'06) received the B.Tech. degree in electrical engineering in 1999 from the Indian Institute of Technology, Chennai, and the M.S. and Ph.D. degrees in electrical engineering from the Polytechnic University, Brooklyn, NY, in 2002 and 2006, respectively.

He is currently with the IntelliTech Microsystems, Inc., Bowie, MD. He is the author or coauthor of more than 50 journal and conference papers. He is also the coauthor of the book *Modeling and Adaptive Nonlinear Control of Electric Motors* (Springer-Verlag 2003). His current research interests include robust and adaptive nonlinear control with applications to electromechanical systems, large-scale systems, and unmanned vehicles.

Farshad Khorrami (S'85–M'88–SM'96) received the Bachelors degrees in mathematics and electrical engineering in 1982 and 1984, respectively, the Master's degree in mathematics and the Ph.D. degree in electrical engineering in 1984 and 1988, respectively, all from The Ohio State University, Columbus.

In September 1988, he joined the Polytechnic University, Brooklyn, NY, as an Assistant Professor, where he is currently a Professor in the Electrical and Computer Engineering Department. His current research interests include control systems with emphasis on nonlinear systems, robotics and automation, unmanned vehicles (fixed-wing and rotary wing aircrafts as well as underwater vehicles and surface ships), smart structures, large-scale systems and decentralized control, adaptive control, and microprocessor-based control and instrumentation. He has also developed educational laboratories in the control systems and robotics area. He is the coauthor of more than 180 refereed journal and conference papers. He is the coauthor of the book *Modeling and Adaptive Nonlinear Control of Electric Motors* (Springer-Verlag, 2003). He is also the holder of 12 U.S. patents on novel smart micropositioners and actuators, control systems, and wireless sensors and actuators. He has developed the Control/Robotics Research Laboratory at the Polytechnic University. His research has been supported by the Army Research Office, the National Science Foundation, Sandia National Laboratory, Office of Naval Research, the Army Research Laboratory, NASA Langley Research Center, and several corporations (e.g., Long Island Lighting Company, Grumman, and IBM).

Prof. Khorrami was the General Chair and program committee member of several international conferences.

David Schoenwald (S'84–M'86–SM'02) received the B.S. degree from the University of Iowa, Ames, in 1986, the M.S. degree from the University of Illinois, Urbana-Champaign, in 1988, and the Ph.D. degree from The Ohio State University, Columbus, in 1992, all in electrical engineering.

Since 1999, he has been a Principal Member of the Technical Staff at Sandia National Laboratories, Albuquerque, NM. From 1992 to 1999, he was a Development Staff Member in the Instrumentation and Controls Division, Oak Ridge National Laboratory, where he was working primarily on industrial control applications. During 1994, he was also an Adjunct Assistant Professor in the Electrical Engineering Department of the University of Tennessee, where he taught a graduate course on nonlinear control systems. His current research interests include agent-based modeling and simulation for logistics and critical infrastructures such as electric power and telecommunications, decentralized control of robotic swarms, and nonlinear control system design.

Dr. Schoenwald has been an Associate Editor of the IEEE Control Systems Society Conference Editorial Board since 1993 and an Associate Editor of the IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY since 2000. He was the recipient of the Martin Marietta Technical Achievement Award and the Industrial Computing Society Courageous User Award in 1995 and 1996, respectively, for his work on a direct-drive industrial sewing machine. From 1996 to 2000, he was an Associate Editor for the IEEE CONTROL SYSTEMS MAGAZINE, for which he guest edited a special issue on autonomous vehicles in December 2000. He was both an elected and appointed member of the CSS Board of Governors from 2003 to 2006.